



Attorney Docket No.: P17412

Patent

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re the Application of: **Steven K. Reinhardt**

Group Art Unit: **2183**

Application No.: **10/749,618**

Examiner: **Partridge, William B**

Filed: **12/30/2003**

For: **BUFFERING UNCHECKED STORES FOR FAULT DETECTION IN  
REDUNDANT MULTITHREADING SYSTEMS USING SPECULATIVE  
MEMORY SUPPORT**

---

**RCE REPLY**

**Mail Stop RCE**  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir/Madam:

Please withdraw the appeal for which a Notice of Appeal was filed on February 26, 2008 and enter this reply with a request for Continued Examination filed herewith. An appeal brief has not been filed and no decision from the Board of Appeals has been rendered. A four month request for extension of time is being submitted to extend the date for responding from the Notice of Appeal to August 26, 2008. Reconsideration is respectfully requested.

## AMENDMENTS

II. IN THE CLAIMS:

The following Listing of Claims replaces all previous versions.

Listing of Claims

1.-10. (Cancelled).

11. (Currently Amended) A method comprising:

dividing a ~~[[thread that is to be executed]]~~ dynamic sequential program into multiple epochs, wherein each epoch includes 2 or more instructions;

in a redundant multi-threading (RMT) system having leading and trailing threads, redundantly executing in parallel first and second instances for each epoch ~~[[by separately executing corresponding leading and trailing epoch instances]]~~ as the leading and trailing threads in the RMT system;

for the executed first and second epoch instances, saving [[committed results from the executed corresponding leading and trailing epoch instances]] store results as speculative stores to memory, the speculative stores being exposed;

comparing the [[committed results that should correspond to one another from the leading and trailing epoch instances]] exposed stores; and

if they match, committing a single set of [[results]] the exposed stores [[for the executed epoch based on the compared results]].

12. (Currently Amended) The method of claim 11, wherein the speculative stores are from a re-order buffer [[saved results are saved as speculative]].

13. (Previously Presented) The method of claim 12, wherein the two or more instructions executed in the execution of the epoch instances are buffered prior to epoch execution completion.

14. (Cancelled).

15. (New) The method of claim 11, wherein the memory is L1 cache memory.

16. (New) A method, comprising:

breaking a program to be executed into multiple epochs each having two or more instructions;

redundantly executing the program by redundantly executing each epoch separately, and sending speculative results for each epoch to memory;

checking the speculative results for each epoch against each other; and

if they match, committing the results.

17. (New) The method of claim 16, in which the speculative results are speculative stores.

18. (New) The method of claim 16, in which the memory is L1 cache memory.

**REMARKS**

With claim 14 being cancelled and new claim s15-18 added, claims 11-13 and 15-18 are now pending. Based on the claim amendments, the objections and rejections are traversed and reconsideration is respectfully requested.

The Examiner objected to claim 11 because leading and trailing epoch did not have proper antecedent basis. Claim 11 has been amended to overcome this objection. It is worth noting that leading and trailing “epoch” has been changed to leading and trailing “thread” for added clarity with respect to the specification, see, e.g., paragraphs 0040 to 0041.

Claims 11-14 were rejected as indefinite under Section 112 because it was unclear how committed results could be stored in memory and then later “committed.” To redress this rejection and make the claim language more clear, the claims have been appropriately amended.

Finally, the claims are rejected under Section 102 as anticipated by Rotenberg and Reinhardt. Neither reference anticipates the claims because neither teaches or suggests, among other things, dividing a program to be executed into epochs having 2 or more instructions for each epoch, redundantly executing the program by redundantly executing the epochs separately, and then checking their results against each other (and committing them if appropriate) at the epoch boundaries, i.e., checking and committing the results epoch by epoch, rather than simply program instruction by program instruction.

Accordingly, a Notice of Allowance is respectfully requested.

**CONCLUSION**

All of the claims are in condition for allowance. Accordingly, Applicant respectfully request the Examiner to pass this case to issue at the Examiner's earliest possible convenience.

If the Examiner believes, for any reason, that personal communication will expedite prosecution of this application, the Examiner is invited to telephone the undersigned at 512/238-7253.

Respectfully submitted,

Date: August 26, 2008

/Erik Nordstrom, Reg. No. 39,792/

Erik R. Nordstrom  
Registration No. 39,792